

Pembuatan Aplikasi “Petra Then and Now” tentang Semua Kegiatan di Universitas Kristen Petra

Dian Christiani Linggadihardja¹, Andreas Handojo², Adi Wibowo³

Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra

Jl. Siwalankerto 121 – 131 Surabaya 60236

Telp. (031) – 2983455, Fax. (031) – 8417658

E-mail: dian_cl@yahoo.com¹, handojo@petra.ac.id², adiw@petra.ac.id³

ABSTRAK

Di era yang maju seperti saat ini, teknologi menjadi bagian yang tidak dapat dilepaskan. Kecepatan dalam mencari sebuah informasi menjadi krusial dan cara lama pun mulai ditinggalkan. Tidak heran bila internet menjadi salah satu kebutuhan yang penting, dan penggunaan *web* untuk menjadi sarana pencari informasi sudah tidak asing lagi.

Penelitian ini dimaksudkan untuk memudahkan pengguna mencari informasi mengenai kegiatan yang ada di Universitas Kristen Petra. Dengan menggunakan *web* saja pengguna dapat melihat daftar kegiatan yang ada di Universitas Kristen Petra, tidak hanya kegiatan yang sedang berlangsung saat ini ataupun yang akan datang (dikemas dalam tampilan kalender), namun juga kegiatan yang sudah lampau dan bersifat historikal (dikemas dalam bentuk *timeline*).

Selain dimudahkan dalam pencarian informasi mengenai kegiatan yang ada, pengguna juga dapat melihat dokumentasi kegiatan bersejarah di Universitas Kristen Petra sehingga dapat mengingat kejadian-kejadian penting yang sudah lewat namun pernah terjadi.

Kata Kunci: Kegiatan, Jadwal, Sejarah, *Website*, *Google Calendar*.

ABSTRACT

In this well developed era, technology becomes a part that can not be neglected. Speed in searching for information has become crucial and the old ways are gradually left behind. It is not surprising that the internet takes an important role in life; website has become a common media to look for information.

This thesis is meant to ease the users who look for information about events in Petra Christian University. By using web, users can see list of events, not only events that are happening or upcoming (displayed as a calendar), but also past and historical events (displayed as a timeline) in Petra Christian University.

Besides being facilitated in looking for event information, users also can see documentation of historical events and be reminded of important things that had already happened.

Keywords: Event, Schedule, History, Website, Google Calendar.

1. PENDAHULUAN

Universitas Kristen Petra sudah berdiri sejak tahun 1961 dan sudah banyak sekali kegiatan yang dilaksanakan dan dapat menjadi pelajaran untuk ke depannya. Berbagai macam kegiatan telah berlangsung dan bertambah banyak seiring bertumbuhnya Universitas Kristen Petra. Kegiatan bersejarah

di Universitas Kristen Petra sering hanya terbengkalai dan terlupakan setelah kegiatan tersebut dilakukan. Tidak adanya fasilitas yang memadai untuk menampung dan mengarsip dengan baik sehingga jarang sekali orang yang mengetahui tentang sejarah Universitas Kristen Petra. Berdasarkan *survey* yang dilakukan dengan sumber Ketua Departemen Matakuliah Umum, adapun tempat penyimpanan yang bersifat *digital* telah dibuat namun kurang mendukung, dan dalam *website* tempat penyimpanan, yaitu *Petra Chronicle*, kegiatan tersebut diarsipkan berdasarkan kategori yang tidak tematik sehingga informasi yang disediakan kurang bisa diterima dengan baik. Disamping itu, sisi publikasi tentang sejarah Universitas Kristen Petra juga kurang menonjol. Banyaknya kegiatan terkadang tidak dapat ditampung dalam satu wadah fisik dan harus dibagi di beberapa tempat sesuai jenis kegiatan. Dengan dibaginya tempat untuk setiap jenis kegiatan, mahasiswa harus pergi ke lokasi tertentu untuk melihat kegiatan apa saja yang sedang ada atau yang akan diadakan. Hal tersebut diatas membuang waktu yang seharusnya dapat dihemat.

Sebuah wadah yang dapat menampung seluruh kegiatan dan dapat diakses dimana saja sangat dibutuhkan. Wadah tersebut juga sebaiknya dapat menampilkan seluruh kegiatan ataupun kegiatan tertentu saja dan dapat menampilkan kegiatan yang sudah lewat beserta foto-foto atau dokumen yang bersangkutan dengan kegiatan tersebut.

Dengan dibuatnya *Petra Then and Now* akan memudahkan mahasiswa untuk melihat seluruh kegiatan yang ada, baik kegiatan yang sudah berjalan dan kegiatan yang akan dijalankan. Selain itu dengan adanya aplikasi ini maka akan menjembatani *Petra Chronicle* dengan kegiatan lain baik yang sudah terjadi maupun yang akan terjadi sehingga menjadi informasi yang dapat diterima dengan baik. Dengan fasilitas kalender yang dapat diunduh akan memudahkan publikasi untuk setiap kegiatan karena informasi kegiatan mengenai tempat atau waktu pelaksanaan akan jelas.

2. DASAR TEORI

2.1. MySQL database

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis. [3] Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya.

SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.[5]. Database MySQL sendiri memiliki *syntax* yang cukup berbeda dari segi format dibandingkan dengan Yii.

2.2. User Group

User group adalah ungkapan yang digunakan untuk mewakili beberapa pengguna. Setiap pengguna memiliki hak akses yang sama namun tidak saling berbenturan. Pengguna diantaranya adalah Program Studi, Unit Kegiatan Mahasiswa, Unit di Universitas Kristen Petra, Organisasi Mahasiswa, dan Universitas itu sendiri. Hak akses yang sama berarti sama-sama dapat menginput, mengedit, dan menghapus kegiatan, sedangkan tidak saling berbenturan maksudnya adalah setiap kegiatan yang dimiliki oleh setiap *user* tidak dapat dihapus ataupun diedit oleh *user* lain.

2.3. Yii Framework

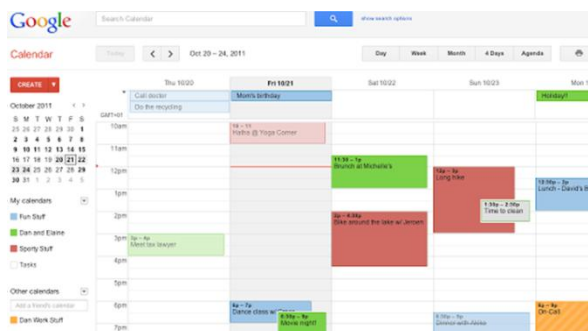
Yii adalah *framework* (kerangka kerja) PHP berbasis-komponen, berkinerja tinggi untuk pengembangan aplikasi Web 2.0. [4] Yii menyediakan *reusability* maksimum dalam pemrograman Web dan mampu meningkatkan kecepatan pengembangan secara signifikan.

Untuk menjalankan aplikasi Web berbasis Yii, diperlukan *Web server* yang mendukung PHP 5.1.0. Bagi para pengembang yang ingin menggunakan Yii, mengerti pemrograman berorientasi-objek (OOP) akan sangat membantu, karena Yii merupakan *framework* OOP murni [7]. Tahap pertama untuk menggunakan Yii adalah mengkonfigurasi *main configuration*. Konfigurasi ini digunakan untuk memberi alamat, menentukan nama aplikasi, mengatur kata sandi untuk *yii*. *GII* adalah fungsi dari *yii* yang dapat membuat *object model* dari database.

2.4. Google Calendar

Google Calendar adalah salah satu fasilitas dari *Google* yang dapat digunakan pada saat akan membuat suatu acara. Dengan menuliskan nama acara dan rinciannya, *Google* akan mengirimkannya sebagai undangan ke beberapa *e-mail* teman. Apabila banyak acara, *Google Calendar* dapat memberikan pengingat pada *e-mail* maupun pada *handphone* sesuai waktu yang kita tentukan.

Google Calendar merupakan salah satu layanan yang diberikan oleh *Google* sebagai pengingat. Layanan ini dapat disinkronkan dengan *smartphone*. Sehingga tidak perlu takut untuk lupa agenda karena selain disinkronkan, juga dapat diingatkan oleh *Google Calendar* baik dengan *alarm* pada ponsel kita, dengan SMS, maupun *email*[5]. Tampilan *Google Calendar* dapat dilihat pada Gambar 1.



Gambar 1. Google Calendar [2]

2.5. Google Calendar API

Google Calendar API memungkinkan sebuah program untuk melakukan banyak operasi yang disediakan oleh *Google Calendar*. Dengan menggunakan API ini, kegiatan-kegiatan yang ada di kalender dapat ditampilkan secara umum. Sesi yang telah dikonfirmasi dapat mengakses kalender yang bersifat pribadi dan membuat, mengganti serta menghapus kegiatan dalam kalender yang menampung kegiatan-kegiatan tersebut.

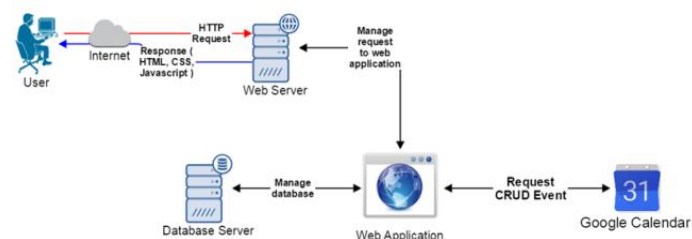
Situs atau aplikasi yang ingin berintegrasi lebih dalam dengan *Google Calendar* dapat memanfaatkan *Google Calendar API*. Integrasi yang dibuat oleh sebuah *web application* dapat membuat dan menampilkan isi kalender. *Desktop application* bersinkronisasi dengan kalender pengguna dengan *desktop application* yang ada [1]. *Google API* untuk mengambil data *client* serta melakukan *authentication* menggunakan *access token*. terdapat *client secret* dan *client id* yang mengindikasikan pengguna tertentu.

3. DESAIN SISTEM

Dalam skripsi ini, terdapat aplikasi untuk *guest* dan *user group* yang disebut dengan *Petra Then and Now* dan aplikasi untuk administrator yang disebut dengan *control panel*. Kedua aplikasi tersebut dibuat berbasis *web* dengan menggunakan bahasa pemrograman HTML 5, PHP, dan Javascript.

Aplikasi untuk *guest* hanya dapat digunakan untuk melihat *event* historikal dan *event* yang ada di *Google Calendar* pada *website*. Aplikasi untuk *user group* dapat digunakan selain untuk melihat *event-event* yang ada, dapat juga digunakan untuk membuat *event*, mengedit *event*, dan menghapus *event* melalui *request* ke administrator. Semua fitur tersebut akan melibatkan *web application*, *database server*, dan *web server*. Data yang diperoleh dan digunakan aplikasi disimpan di *database server* berupa *Google Calendar* dan *database MySQL* yang disimpan dalam *server*.

Sedangkan aplikasi untuk administrator (*control panel*) utamanya digunakan administrator untuk mempermudah dalam mengelola aplikasi *Petra Then and Now* ini. Untuk mendukung fungsi tersebut, aplikasi ini menyediakan fitur-fitur dasar seperti master (*event* dan *user*), dan validasi *event*. Semua fitur tersebut melibatkan komponen-komponen yang sama seperti pada aplikasi untuk *user group*. Gambaran alur sistem ini dapat dilihat pada Gambar 2.



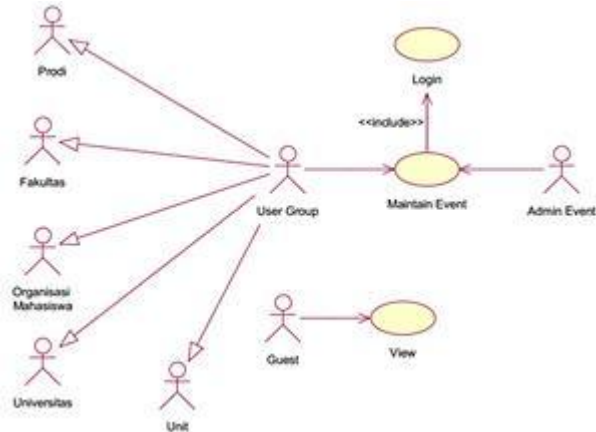
Gambar 2. Garis besar sistem aplikasi *Petra Then and Now*

3.1. Use Case Diagram

Use case diagram pada Gambar 3 menjelaskan aktivitas pada aplikasi *Petra Then and Now* termasuk aplikasi *guest* dan *user group*. Berikut penjelasan mengenai *use case diagram*:

- Pada *use case View*, *guest* hanya dapat melihat *event* yang ada dalam bentuk *Google Calendar* beserta halaman deskripsinya.

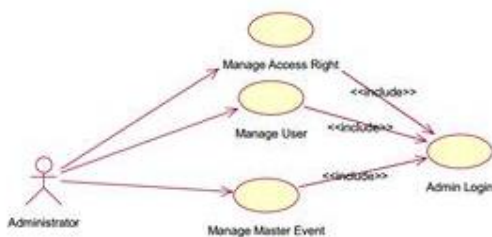
- Pada *use case Login*, *user group* yang sudah memiliki akun dapat melakukan *login* ke dalam aplikasi.
- Pada *use case Maintain Event*, *user group* yang telah *login*, dapat melakukan fitur-fitur yang disediakan diantaranya adalah *create event*, *edit event*, dan *delete event*. Semua fitur tersebut membutuhkan verifikasi dari *admin event*.
- Pada *use case Admin Login*, hanya administrator yang dapat melakukan *login* karena halamannya pun akan berbeda dengan halaman *user group*.



Gambar 3. Use case diagram aplikasi Petra Then and Now

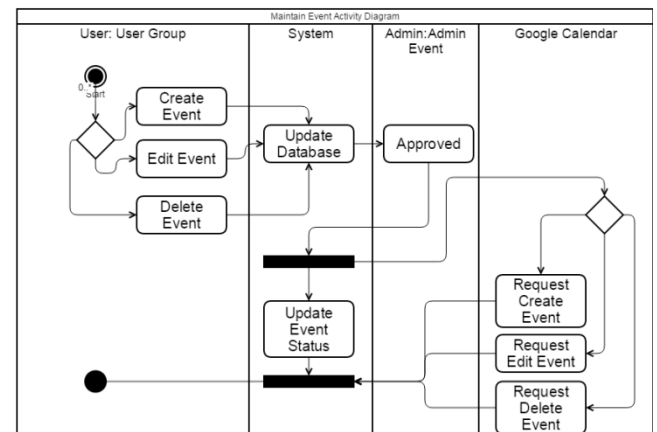
Use case diagram pada Gambar 4 menjelaskan aktivitas pada *control panel*. Berikut penjelasan mengenai *use case diagram* pada *control panel*:

- Pada *use case Manage Access Right*, administrator dapat melakukan CRUD (*create, read, update, delete*) jenis *user* yang ada.
- Pada *use case Manage User*, administrator dapat melakukan CRUD (*create, read, update, delete*) untuk *user* yang ada.
- Pada *use case Manage Master Event*, administrator dapat melakukan CRUD (*create, read, update, delete*) untuk *event* yang ada.



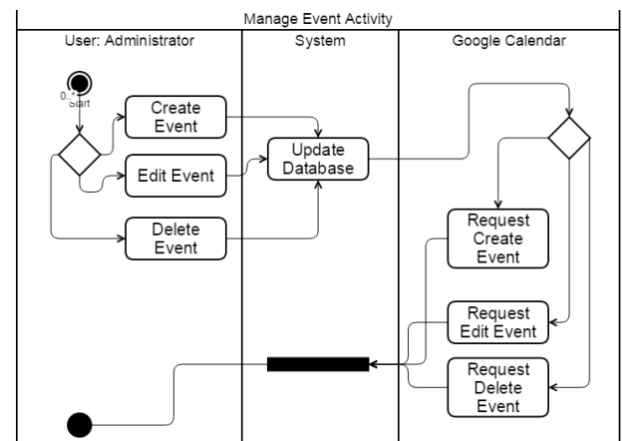
Gambar 4. Use case diagram aplikasi Petra Then and Now untuk Administrator

3.2. Activity Diagram



Gambar 5. Maintain Event Activity Diagram untuk user group

Gambar 5 menjelaskan bahwa setelah *user group* melakukan *login*, maka *user group* akan dapat melakukan fitur *create event*, *edit event*, dan *delete event* yang nantinya dikirim ke *database*. Setelah itu administrator dapat melakukan validasi dari setiap *event*. *Event* yang sudah divalidasi akan langsung masuk ke *Google Calendar* dan dapat ditampilkan di sana.

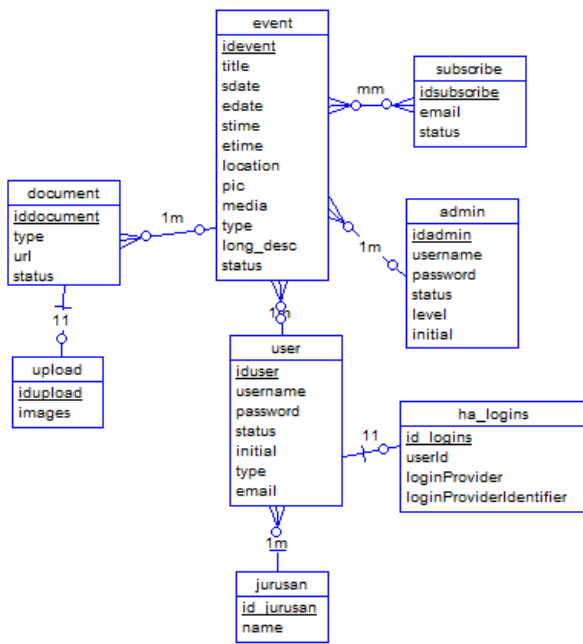


Gambar 6. Manage Master Event Activity Diagram untuk administrator

Gambar 6 menjelaskan bahwa administrator dapat melakukan CRUD (*create, read, update, delete*) untuk *event* yang ada yang langsung masuk ke *Google Calendar*. Administrator juga dapat melakukan CRUD (*create, read, update, delete*) pada *manage user activity diagram* dan *manage documentation activity diagram*.

3.3. Entity Relationship Diagram (ERD)

Untuk menunjukkan desain database, diperlukan *Entity Relationship Diagram* untuk mendesain *entity* dan *attribute* yang dibutuhkan sistem serta *relationship* antar *entity* tersebut. *Entity Relationship Diagram* dapat dilihat pada Gambar 7.



Gambar 7. Entity Relationship Diagram

4. IMPLEMENTASI SISTEM

4.1. Implementasi Awal

Aplikasi yang telah dibuat merupakan sebuah *web application* yang dibuat dengan menggunakan PHP dan JavaScript. Untuk aplikasi ini dapat dijalankan dengan lancar versi minimal PHP >= 5.1 dibutuhkan untuk Yii Framework.

4.2. Pembuatan Fungsi

Pada bab ini, akan dibahas fungsi-fungsi yang digunakan oleh fitur-fitur aplikasi pada desain yang telah dijelaskan pada bab sebelumnya. Berikut merupakan daftar fungsi utama yang digunakan dalam pembuatan aplikasi skripsi ini yang dapat dilihat pada uraian dibawah ini:

4.2.1. Google Calendar

Google Calendar digunakan sebagai tempat penyimpanan data selain di database server. Data yang disimpan berupa judul, tanggal, dan lokasi. Pada yii, ada hal yang dilakukan agar dapat melakukan fungsi-fungsi Google Calendar API. Pada Gambar 8 adalah cara agar dapat menggunakan fungsi Google Calendar. Fungsi `require_once` tidak dapat dipanggil tanpa diapit dengan `spl_autoload`. Hal ini disebabkan fungsi dari `autoload` yii bertabrakan dengan fungsi `autoload` dari Google Calendar.

```

spl_autoload_unregister(array('YiiBase','autoload'));
require_once(Yii::app()->basePath.'\extensions\gcal\insert_event.php');
spl_autoload_register(array('YiiBase','autoload'));

```

Gambar 8. Kutipan code yang digunakan untuk dapat memanggil fungsi Google Calendar

4.2.2. Insert Event

Insert event pada *Google Calendar* menggunakan fungsi API yang disediakan. Pada Gambar 9 dapat dilihat fungsi `setsummary`, `setlocation`, `setdatetime`, `setstart`, dan `setend`

sebagai parameter untuk memasukkan *event* ke *Google Calendar*. Fungsi ini menghasilkan `event_id` yang digunakan untuk melakukan fungsi *update* dan *delete*.

```

$event = new
Google_Service_Calendar_Event();
$event->setSummary($event_title);
$event->setLocation($event_location);
$event-
>setDescription('http://localhost/tugas/event/guest/'.$id);
$start = new
Google_Service_Calendar_EventDateTime();
$start->setDateTime($start_event);
$start->setTimeZone('Asia/Jakarta');
$event->setStart($start);
$end = new
Google_Service_Calendar_EventDateTime();
$end->setDateTime($end_event);
$end->setTimeZone('Asia/Jakarta');
$event->setEnd($end);
$insertEvent = $service->events-
>insert($calendarId, $event);
return $insertEvent->getId();
}
else{
$htmlBody .= '<p>Problems creating the
client</p>';
}
}

```

Gambar 9. Code untuk melakukan insert event

4.2.3. Event (User Group)

Ada beberapa fitur yang disediakan untuk *user group* yaitu fitur *create event*, *update event*, *delete event*, *create documentation* dan *update documentation*. Pada Gambar 10 adalah fungsi untuk *create event* untuk *user group*.

```

public function actionCreate()
{
$model=new Event;
if(isset($_POST['Event']))
{
$model->attributes=$_POST['Event'];
$model->status='0';
$model->iduser=Yii::app()->user->id;
$model->isdelete='0';
$model->idadmin='0';
if($model->save())
$this-
>redirect(array('view','id'=>$model->id));
}
$this->render('create',array(
'model'=>$model,
));
}

```

Gambar 10. Fungsi *create event* pada aplikasi

Pada Gambar 11 adalah fungsi untuk membuat *documentation* untuk *user group*.

```

public function actionCreate(){
$model=new Document;
if(isset($_POST['Document'])) {
$model->attributes=$_POST['Document'];
$model->iduser=Yii::app()->user->id;
$model->status='0';
if($model->save())
$this->redirect(array('view','id'=>$model-
>iddocument));
}
}

```

Gambar 11. Fungsi *create documentation* pada aplikasi

4.2.4. Event (Admin)

Admin dapat melakukan fungsi dasar seperti pada *user group*. Ada beberapa fitur tambahan yang dapat dilakukan admin diantaranya adalah melakukan validasi untuk kegiatan seperti yang terlihat pada Gambar 12.

```
public function actionApprove(){
Yii::import('booster.components.TbEditableSaver');
$es = new TbEditableSaver('Event');
$es->update();
$lala= Event::model()-
>findAll(array('condition'=>'id
=:cekid','params'=>array(':cekid'=>$es-
>model->id)));
foreach ($lala as $result):
$result->idadmin;
endforeach;
$result->idadmin=Yii::app()->user->id;
$result->update();
}
```

Gambar 12 Validasi kegiatan

Dokumentasi yang sudah diinputkan oleh user perlu di validasi oleh admin. Berikut pada Gambar 13 adalah fungsi validasi dokumentasi.

```
public function actionApprove(){
Yii::import('booster.components.TbEditableSaver');
$es = new TbEditableSaver('Document');
$es->update();
$lala= Document::model()-
>findAll(array('condition'=>'iddocument
=:cekid','params'=>array(':cekid'=>$es-
>model->iddocument)));
foreach ($lala as $result):
$result->idadmin;
endforeach;
$result->idadmin=Yii::app()->user->id;
$result->update();
}
```

Gambar 13. Validasi dokumentasi

Sebuah kegiatan ada yang bersifat historikal, pada Gambar 14 adalah fungsi yang membuat kegiatan menjadi bersifat historikal.

```
public function actionHistorical(){
Yii::import('booster.components.TbEditableSaver');
$es = new TbEditableSaver('Event');
$es->update();
$lala= Event::model()-
>findAll(array('condition'=>'id
=:cekid','params'=>array(':cekid'=>$es-
>model->id)));
foreach ($lala as $result):
$result->idadmin;
endforeach;
$result->idadmin=Yii::app()->user->id;
$result->update();
}
```

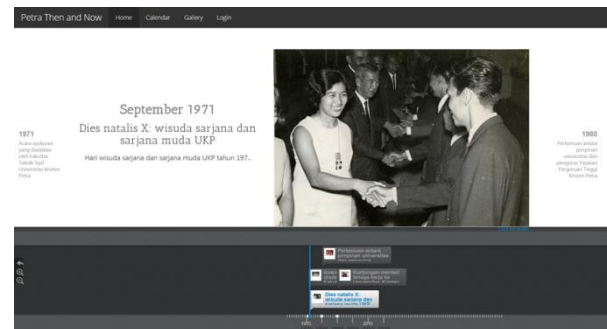
Gambar 14. Membuat status event historikal

5. PENGUJIAN SISTEM

Pada bab ini dibahas pengujian sistem yang diimpelentasikan pada skripsi ini. Pengujian dilakukan berdasarkan hak akses pengguna yang terbagi atas 4 jenis yang terdiri dari *Guest*, *User group*, *Admin Event* dan *Head Admin*.

5.1. Pengujian Aplikasi untuk Guest

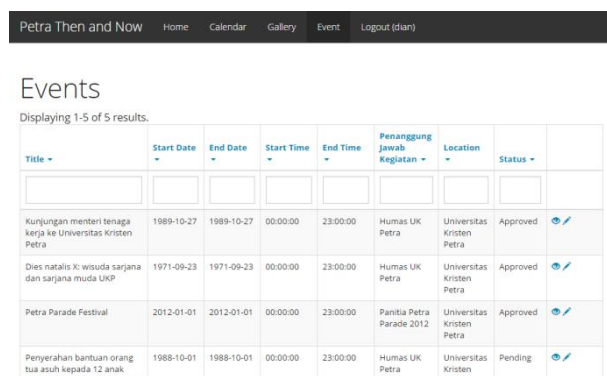
Pada bagian ini akan dibahas alur akses *website* dari sisi *guest*. Sebagai *guest*, tidak diperlukan *login*, tetapi hak akses dibatasi hanya pada melihat *event* dan dokumentasi. Berikut pada Gambar 15 adalah tampilan home untuk *Guest*.



Gambar 15. Tampilan Home

5.2. Pengujian Aplikasi untuk Member

Pada bagian ini dibahas fitur-fitur yang dapat diakses setelah berhasil *login* sebagai *user group*. Fitur tambahan bagi pengguna berstatus *User Group* adalah halaman “Event”. Sementara hak akses *user group* lebih luas yaitu dapat melakukan operasional pada *event* termasuk *create event*, *update event*, *delete event*, *create event documentation* dan *update event documentation*. Pada Gambar 16 dapat dilihat tampilan daftar *events* yang telah dibuat oleh *user group*.



Gambar 16. Tampilan Manage Event untuk User Group

5.3. Pengujian Aplikasi untuk Admin Event

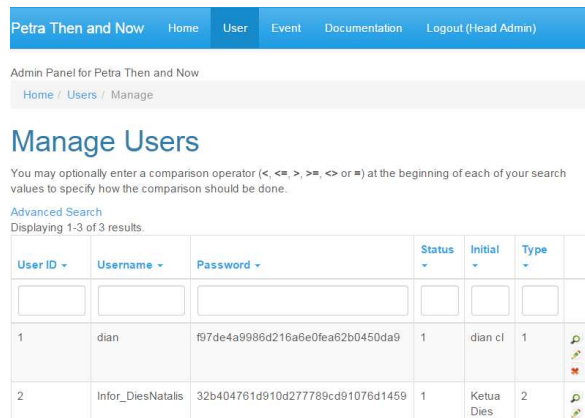
Pada pengujian aplikasi untuk bagian *admin event* secara garis besar *admin event* bertugas untuk melakukan validasi kegiatan dan dokumentasi yang dibuat oleh *user group*. Admin *Event* sendiri juga dapat melakukan operasional yang dilakukan oleh *user group*. Pada Gambar 17 dapat dilihat *admin event* dapat melakukan validasi *event* dan mengganti status historikal *event*.



Gambar 17. Halaman manage event admin event

5.4. Pengujian Aplikasi untuk Head Admin

Pada pengujian aplikasi untuk bagian *head admin* secara garis besar *head admin* dapat melakukan semua operasi seperti *admin event*. Namun *head admin* bersifat lebih memiliki otoritas dalam hal pengaturan *user*. Pada Gambar 18 dapat dilihat pengaturan untuk *user* khusus untuk Head Admin.



Gambar 18. Tampilan manage user head admin

6. KESIMPULAN

Berdasarkan hasil pengujian dapat disimpulkan beberapa hal sebagai berikut:

1. Kegiatan dapat ditampilkan di *Google Calendar* tanpa harus login, sebagai guest dapat mengunduh dan menyalin kegiatan tersebut ke kalender pribadi.
2. Kegiatan bersejarah di Universitas Kristen Petra dapat ditampilkan melalui *timeline* yang berada pada home. Kegiatan tersebut ditampilkan berbentuk rangkaian gambar dengan urutan waktu dari mulai yang paling awal hingga paling baru.
3. Pengkategorian kegiatan berdasarkan tipe dari pembuat kegiatan, seperti misalnya kegiatan Himainfra akan berwarna merah muda sedangkan kegiatan Himahotra akan berwarna hijau.

4. Setiap pengguna memiliki hak akses untuk kegiatan yang dibuatnya tanpa mengganggu kegiatan lain meskipun sama-sama berasal dari fakultas yang sama.
5. Informasi mengenai sejarah Universitas Kristen Petra didapatkan melalui akses dari website perpustakaan Universitas Kristen Petra.
6. Pengujian menggunakan *Google Chrome* dan *Mozilla Firefox* dari beberapa sampel menunjukkan aplikasi ini bekerja baik untuk kedua *browser*.

7. REFERENSI

- [1] Google. 2015. *Google Calendar API*. URL: <https://dev.mysql.com/doc/refman/4.1/en/what-is-mysql.html>
- [2] Google. *Google Calendar*. URL: <https://www.google.com/calendar>
- [3] Indosite. 2014. *Indosite Pengertian MySQL*. URL: www.indosite.com/web/tutorials/pengertian-mysql/
- [4] Makarow, A. 2011. *Yii Framework*. URL: www.yiiframework.com
- [5] Oracle. 2015. *What is mysql*. URL: <http://www.oracle.com/technetwork/database/mysql/index.html>
- [6] Wahono, R. S. 2013. *Menggunakan Google Kalender*. URL: <http://ilmukomputer.org/2013/06/10/menggunakan-google-kalender/>
- [7] Yii Software LLC. 2015. *yiiframework*. URL: <http://www.yiiframework.com/>